

BAB II

LANDASAN TEORI

2.1 Distribusi

Distribusi adalah suatu kegiatan untuk memindahkan barang dari pihak supplier kepada pihak agen dalam suatu *supply chain*. Distribusi merupakan suatu kunci dari keuntungan yang akan diperoleh perusahaan karena distribusi secara langsung akan mempengaruhi biaya dari *supply chain* dan kebutuhan agen. Jaringan distribusi yang tepat dapat digunakan untuk mencapai berbagai macam tujuan dari *supply chain*, mulai dari biaya yang rendah sampai respon yang tinggi terhadap permintaan agen (Chopra & Meindl, 2010 : 86).

Pada pendistribusian barang kepada konsumen, Swastha (1984), menyebutkan terdapat lima klasifikasi sebagai berikut :

1. Produsen – Konsumen

Merupakan saluran distribusi langsung karena memungkinkan produsen untuk langsung dapat menjual barang kepada konsumen akhir.

2. Produsen – Pengecer – Konsumen

Saluran distribusi ini dapat disebut dengan saluran distribusi langsung. Pengecer melakukan pembelian barang kepada produsen. Adapula beberapa produsen yang mendirikan toko pengecer sehingga dapat secara langsung melayani konsumen.

3. Produsen – Pedagang besar – Pengecer – Konsumen

Saluran distribusi semacam ini banyak digunakan oleh produsen dan dinamakan sebagai saluran distribusi tradisional. Dalam saluran distribusi ini produsen hanya melayani penjualan dalam jumlah yang besar kepada pedagang besar saja. Pembelian oleh pengecer dilayani oleh pedagang besar dan pembelian oleh konsumen dilayani oleh pengecer saja.

4. Produsen – Agen – Pengecer – Konsumen

Dalam saluran distribusi ini, produsen memilih agen (agen penjualan atau agen pabrik) sebagai penyalurnya. Agen menjalankan perdagangan besar

dalam saluran distribusi yang ada. Sasaran penjualannya ditujukan kepada pengecer besar.

5. Produsen – Agen – Pedagang besar – Pengecer – Konsumen

Dalam saluran distribusi ini, produsen sering menggunakan agen sebagai perantara untuk menyalurkan barangnya kepada pedagang besar yang kemudian menjualnya kepada toko – toko kecil. Agen yang terlibat adalah agen penjualan.

2.2 Vehicle Routing Problem

2.2.1 Definisi Vehicle Routing Problem

Vehicle Routing Problem (VRP) merupakan permasalahan utama dalam manajemen distribusi. Perusahaan-perusahaan yang bergerak di bidang distribusi akan selalu menghadapi permasalahan ini dalam melaksanakan kegiatan operasionalnya. Permasalahan distribusi ini dapat berupa distribusi barang ataupun manusia. Permasalahan distribusi barang sering dihadapi oleh perusahaan-perusahaan manufaktur dalam mengirimkan produknya kepada konsumen. Sedangkan untuk permasalahan distribusi manusia biasanya dihadapi oleh perusahaan yang bergerak di bidang jasa, misalnya transportasi bagi penyandang cacat dan permasalahan penumpang bis kota.

Menurut Rahmi dan Murti (2013): *Vehicle Routing Problem* (VRP) merupakan permasalahan dalam sistem distribusi yang bertujuan untuk membuat suatu rute yang optimal, dengan sekelompok kendaraan yang sudah diketahui kapasitasnya, agar dapat memenuhi permintaan konsumen dengan lokasi dan jumlah permintaan yang telah diketahui. Suatu rute yang optimal adalah rute yang memenuhi berbagai kendala operasional, yaitu memiliki total jarak dan waktu perjalanan yang ditempuh terpendek dalam memenuhi permintaan konsumen serta menggunakan kendaraan dalam jumlah yang terbatas. Berikut ini adalah beberapa kendala atau batasan yang harus dipenuhi dalam VRP yaitu:

1. Rute kendaraan dimulai dari depot dan berakhir di depot,

2. Masing-masing konsumen harus dikunjungi sekali dengan satu kendaraan,
3. Kendaraan yang digunakan adalah homogen dengan kapasitas tertentu, sehingga permintaan konsumen pada setiap rute yang dilalui tidak boleh melebihi kapasitas kendaraan.
4. Jika kapasitas kendaraan sudah mencapai batas, maka konsumen berikutnya akan dilayani oleh shift berikutnya.

2.2.2 Klasifikasi Jenis-jenis VRP

Berdasarkan batasan-batasan yang dipertimbangkan sesuai dengan kondisi nyata, VRP dibagi menjadi 8 tipe yaitu :

1. *Capacitated Vehicle Routing Problem (CVRP)*. Ciri khas dari permasalahan ini adalah sistem distribusi memiliki satu depot dan hanya memiliki batasan kapasitas kendaraan dengan fungsi tujuan meminimalkan total biaya transportasi. Contoh nyata dari permasalahan ini adalah distribusi semen dimana pabrik mengirimkan semennya ke gudang penyangga setiap hari. Contoh lainnya adalah pengumpulan sampah dari satu wilayah ke wilayah lain oleh petugas sampah.
2. *Distance Constrained Vehicle Routing Problem (DCVRP)*. Tipe permasalahan ini merupakan turunan dari permasalahan CVRP, dengan menambahkan batasan total waktu tempuh dari setiap rute. Tujuan dari permasalahan ini adalah meminimumkan total jarak atau waktu tempuh.
3. *Vehicle Routing Problem with Back Hauls (VRPB)*. Ciri khas dari permasalahan ini adalah CVRP yang dibagi dalam 2 tipe konsumen, yaitu konsumen yang meminta layanan angkut (*back haul*) dan konsumen yang meminta layanan antar (*line haul*). Dimana sebelum melakukan pengangkutan maka kendaraan harus memenuhi semua jadwal pengiriman terlebih dahulu.
4. *Vehicle Routing Problem with Time Windows (VRPTW)*. VRP yang mempertimbangkan batasan-batasan *time windows* (dimana tiap konsumen memiliki waktu kunjungan tertentu).
5. *Vehicle Routing Problem with Pick Up and Delivery (VRPPD)*. VRP dimana setiap customer memiliki satu lokasi *pick up* (jemput) dan satu

lokasi *delivery* (antar). Dengan lokasi *pick up* bisa identik bisa tidak dengan lokasi *delivery*.

6. *Vehicle Routing Problem with Back Hauls and Time Windows (VRPBTW)*. VRPB yang memperhatikan *time windows*.
7. *Vehicle Routing Problem with Pick Up and Delivery with Time Windows (VRPPDTW)*. VRPPD dengan ada aturan kunjungan tertentu pada setiap lokasi penjemputan dan pengiriman.
8. *Multiple Depot Vehicle Routing Problem (MDVRP)*. VRP dengan jumlah depot lebih dari satu.
9. *Open Vehicle Routing Problem (OVRP)*. VRP adalah bahwa kendaraan tidak perlu kembali ke tempat yang merupakan pusat distribusi (depot).

2.2.3 Metode Penyelesaian VRP

Ada 2 prosedur/ metode yang dapat digunakan untuk proses penentuan rute dan penjadwalan transportasi dalam VRP (Hugos, 2003).

- *Saving Matriks Method*
- *The Generalized Assignment Method*

a. Metode Saving Matriks

Saving Matriks merupakan salah satu teknik yang digunakan untuk menjadwalkan sejumlah terbatas kendaraan dari suatu fasilitas dan jumlah kendaraan dalam armada ini dibatasi dan mereka mempunyai kapasitas maksimum yang berlainan. Tujuan dari metode ini adalah untuk memilih penugasan kendaraan dan routing sebaik mungkin. Bowersox. (2002).

Chopra et al. (2003) menerangkan bahwa metode sederhana ini dilakukan dan dapat digunakan untuk memutuskan konsumen ke kendaraan mana, walaupun terdapat kendala waktu atau yang lainnya. Langkah-langkah utama metode *saving matriks* :

- a) Menentukan *matriks* jarak.

Matriks jarak menyatakan jarak diantara tiap pasangan lokasi-lokasi yang akan dikunjungi. Jarak antara lokasi A yang terletak pada koordinat (X_a, Y_a) dan lokasi B yang terletak pada koordinat (X_b, Y_b) dicari dengan menggunakan rumus :

$$\text{Dist}(A,B) = \sqrt{(X_a - X_b)^2 + (Y_a - Y_b)^2}$$

1. Menentukan matriks penghematan (*saving matriks*).

Matriks penghematan menunjukkan penghematan yang terjadi jika menggabungkan 2 konsumen kedalam satu truk. Penghematan bisa dievaluasi berdasarkan jarak atau waktu atau uang.

$S(x,y)$ menyatakan jarak yang dihemat jika perjalanan DC \rightarrow konsumen x dan DC \rightarrow konsumen y dikombinasikan ke sebuah rute perjalanan tunggal yaitu DC \rightarrow konsumen x \rightarrow konsumen y. Rumus untuk mencari besarnya penghematan adalah :

$$S(x,y) = \text{Dist}(DC,x) + \text{Dist}(DC,y) - \text{Dist}(x,y)$$

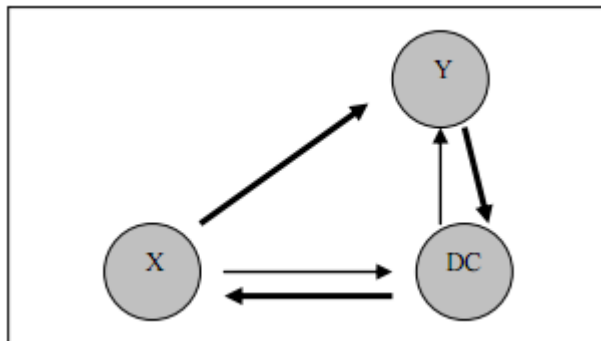
2. Mengalokasikan konsumen-konsumen ke sebuah rute/kendaraan atau menugaskan konsumen pada sebuah rute.

- Pengalokasian konsumen ke sebuah rute/kendaraan harus bisa memaksimalkan penghematan.
- Pencarian solusi dilakukan dengan prosedur iteratif yaitu :
 - a) Pada tahap 1 : tiap konsumen dialokasikan pada truk/rute yang berbeda-beda/terpisah.
 - b) Pada tahap 2 : Dua rute selanjutnya dapat digabungkan pada satu rute/kendaraan dengan didasarkan pada penghematan yang paling tinggi yang bisa diperoleh. Selanjutnya dilakukan pengecekan apakah pengkombinasian tersebut layak atau tidak. Dikatakan layak jika total pengiriman yang harus dilalui melalui rute tersebut tidak melebihi kapasitas kendaraan.

3. Menentukan urutan konsumen/urutan pengiriman pada sebuah rute. Tujuan dari tahap ini adalah meminimalkan jarak perjalanan yang harus ditempuh tiap kendaraan. Untuk mendapatkan rute pengiriman yang optimal dilakukan dalam 2 tahap :

- (1) Menentukan rute pengiriman awal untuk tiap kendaraan dengan menggunakan prosedur *Farthest insert/Nearest insert/dll*.

(2) Melakukan perbaikan dengan menggunakan prosedur tertentu.



Gambar 2.1 Diagram Matriks Penghematan (Chopra et al. 2002)

Keterangan :

1. Jarak adalah Euclidean atau Rectilinear
 2. Penyimpanan atas konsolidasi dua pelanan X,Y
 - Eliminasi dimulai dari X sampai Dc dan
 - Eliminasi dimulai dari DC sampai Y
 - Tapi penambahan dimulai dari X sampai Y
- b. Penjadwalan dan Penentuan Rute Pengiriman dengan *Generalized Assignment*

Langkah-langkah yang dilakukan meliputi :

1. Menentukan *seed point* untuk masing-masing rute/alat angkut dimana *seed point* merupakan pusat perjalanan yang diambil oleh tiap alat angkut. Prosedurnya adalah :

- a. Menentukan L_{seed} dengan rumus :

$$L_{seed} = \text{total permintaan} / \text{jumlah alat angkut}$$

- b. Dimulai dari pelanggan manapun, lakukan gerakan searah jarum jam dimulai dari DC untuk mendapatkan *cone* yang dialokasikan untuk tiap *seed point* sesuai dengan besarnya L_{seed} . Langkah-langkah untuk mendapatkan *cone*, yaitu :

- Menentukan posisi sudut angular (θ_i) dari masing-masing pelanggan i yang memiliki koordinat (x_i, y_i).

- $\theta_i = \tan^{-1} (y_i / x_i)$

- Menggeser searah jarum jam untuk menentukan urutan pelanggan yang akan masuk dalam *cone* berdasarkan permintaan pelanggan dan L_{seed} .
- c. Pada tiap *cone*, *seed point* diletakkan di tengah *cone* dan koordinat *seed point* i dihitung dengan rumus :

$$X_i = d_i \times \cos(\alpha)$$

$$Y_i = d_i \times \sin(\alpha)$$

Dimana :

d_i = Jarak antara pelanggan yang paling jauh dari *cone* dengan DC.

α = Sudut yang di bentuk dari penjumlahan sudut pelanggan awal dengan sudut pelanggan terjauh yang masuk *cone* dan selanjutnya dibagi 2 .

- d. Pembentukan *cone* kedua dan selanjutnya di mulai dari sudut hasil penyisipan pelanggan terakhir pada *cone* sebelumnya.
2. Mengevaluasi biaya penyisipan untuk masing-masing pelanggan
- Untuk tiap *seed point* S_k dan pelanggan i , biaya penyisipan merupakan jarak tambahan yang akan di tempuh jika pelanggan disisipkan pada sebuah perjalanan dari DC ke *seed point*.

$$c_{ik} = \text{Dist}(DC, i) + \text{Dist}(i, S_k) - \text{Dist}(DC, S_k)$$

3. Menugaskan/mengalokasikan masing-masing pelanggan pada tiap kendaraan/rute

Penugasan pelanggan pada kendaraan/rute diformulasikan menggunakan *integer programming* dengan fungsi tujuan meminimalkan biaya penyisipan dan fungsi batasan kapasitas alat angkut.

Variabel Keputusan :

$Y_{ik} = 1$, jika pelanggan i dialokasikan ke alat angkut k , 0 jika sebaliknya.

Formulasi *Integer Programing* untuk mengalokasikan pelanggan ke alat angkut

$$\text{Min} = \sum_{k=1}^K \sum_{i=1}^n c_{ik} Y_{ik}$$

Subject to :

$$\sum_{k=1}^K Y_{ik} = 1, \quad i = 1, \dots, n$$

$$\sum_{i=1}^n a_i Y_{ik} \leq b_k, \quad k = 1, \dots, K$$

Dimana :

c_{ik} = biaya penyisipan dari pelanggan i dan *seed point* k

a_i = *Order size* atau permintaan dari pelanggan i

b_k = kapasitas dari alat angkut k

4. Menentukan urutan kunjungan pelanggan untuk tiap kendaraan/rute

Tujuan dari tahap ini adalah meminimalkan jarak perjalanan yang harus di tempuh tiap kendaraan dengan menggunakan metode-metode penentuan urutan kunjungan diantaranya *farthest insert* atau *nearest insert* atau *nearest neighbour*.

2.3 Open Vehicle Routing Problem

OVRP merupakan suatu persoalan manajemen distribusi. Hal yang membedakan OVRP dengan VRP klasik adalah bahwa kendaraan tidak perlu kembali ke tempat yang merupakan pusat distribusi (depot), atau jika dibutuhkan untuk kembali, maka kendaraan tersebut akan kembali dengan mengunjungi rute yang sama seperti pada rute keberangkatannya (Sariklis and Powell, 2000).

Persoalan bahwa kendaraan tidak kembali ke depot akan dihadapi oleh perusahaan yang sama sekali tidak memiliki kendaraan sendiri, atau kendaraannya tidak tepat atau tidak memadai untuk memenuhi permintaan pelanggan. Oleh karena itu, perusahaan terpaksa mengontrak seluruh atau sebagian kendaraan untuk mendistribusikan produknya kepada kurir eksternal (*expeditur*). Kendaraan sewaan akan ditugaskan dalam suatu rute dimana kendaraan tersebut tidak perlu kembali ke depot. Solusi masalah ini akan mengarahkan perusahaan agar memperoleh

jumlah minimum kendaraan yang akan disewa untuk melayani pelanggan dan memberikan sejumlah rute yang meminimalkan biaya perjalanan. Lebih lanjut, pada situasi dimana perusahaan memiliki armada kendaraan sendiri dan permintaan pelanggan yang berubah-ubah secara signifikan sepanjang waktu, solusi akan menyediakan kombinasi solusi yang tepat bagi pemilik atas kendaraan sewaan tersebut.

Secara teoritis, OVRP merupakan suatu persoalan optimisasi kombinatorial NP-hard (*Non Polynomial Hard Problem*), yaitu menyelesaikan OVRP untuk mengoptimalkan path Hamiltonian terbaik pada himpunan pelanggan yang ditugaskan kepada suatu kendaraan. Karena mendapatkan path Hamiltonian terbaik pada himpunan pelanggan merupakan NP-hard, demikian pula dengan OVRP (Brandao, 2004). Hal tersebut menunjukkan kesamaan antara VRP dan OVRP.

2.4 Modifikasi Metode *Clarke-Wright Algoritma*

Clarke-Wright adalah salah satu algoritma heuristik, namun tidak dapat menjamin akan menghasilkan solusi terbaik. Oleh karena itu, diusulkan versi modifikasi dari algoritma *Clarke-Wright* dimana versi paralel CW diimplementasikan karena biasanya menghasilkan hasil yang lebih baik dari algoritma heuristik yang lain (Pichpibul and Kawtummachai, 2013). Pertama matriks jarak Euclidean ($c_{i,j}$) dihitung dengan persamaan berikut:

$$c_{i,j} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \quad (1)$$

Dimana x_i , y_i dan x_j , y_j adalah lokasi geografis pelanggan i dan j . Kedua, total jarak antara pelanggan i dan j dihitung sebagai:

$$S_{i,j} = c_{1,j} - c_{i,j} \quad (2)$$

Dimana $c_{1,j}$ adalah perjalanan jarak antara depot dengan pelanggan j dan $c_{i,j}$ adalah perjalanan jarak antara pelanggan i dan j . Persamaan (2) yang dimodifikasi oleh Bodin et al dari perumusan *Clarke-Wright* yang ditunjukkan dalam persamaan (3). Setelah perhitungan, semua *savings values* yang dikumpulkan dalam *savings list* dihitung dengan persamaan sebagai berikut:

$$s_{i,j} = c_{1,i} + c_{j,1} - c_{i,j} \quad (3)$$

Ketiga, nilai-nilai dalam *savings list* diurutkan dalam *decreasing order*. Selanjutnya, rute yang menggabungkan prosedur dimulai dari atas daftar tabungan (s_i , terbesar). Baik pelanggan i dan j akan digabungkan menjadi rute yang sama jika total permintaan tidak melebihi kapasitas kendaraan dan tidak ada kendala rute eksis. Prosedur rute penggabungan diulang sampai tidak ada penggabungan layak dalam daftar tabungan mungkin. Selanjutnya, dalam hal pelanggan non routed, masing-masing ditugaskan oleh rute yang dimulai di depot, mengunjungi pelanggan yang belum ditetapkan, dan kembali ke depot. CW yang diusulkan adalah pendekatan perbaikan berulang dirancang untuk menemukan solusi optimal global. Telah disajikan dalam empat prosedur yang terdiri dari Clarke-Wright rumus modifikasi, konstruksi terbuka rute, dua-fase seleksi, dan rute postimprovement.

2.4.1 Prosedur Modifikasi Clarke-Wright

Gaskell dan Yellow (Pichpibul and Kawtummachai, 2013) menjelaskan mengenai parameter bentuk rute (λ) yang mengontrol signifikansi relatif dari busur langsung antara dua pelanggan. diusulkan gabungan rumus mereka adalah sebagai berikut:

$$s_{i,j} = c_{1,i} + c_{j,1} - \lambda c_{i,j} \quad (4)$$

Berdasarkan persamaan (2), persamaan (4) juga dimodifikasi dengan menghapus $c_{j,1}$ untuk pemecahan OVRP dengan persamaan berikut:

$$s_{i,j} = c_{1,j} - \lambda c_{i,j} \quad (5)$$

Parameter yang dapat bervariasi sebagai dipelajari oleh Altinel dan Oncan (2005). Mereka menggunakan pendekatan enumerative sederhana untuk menghasilkan 8820 solusi yang berbeda ($\lambda \in [0,1, 2]$, $\mu \in [0, 2]$, $v \in [0, 2]$). Setelah itu solusi terbaik akan dipilih. Dalam memesan untuk menghindari solusi berulang dan memakan waktu, diterapkan hanya parameter tunggal (λ) untuk prosedur ini.

2.4.2 Prosedur Konstruksi Rute Terbuka

Untuk memecahkan CVRP dengan Algoritma CW, setelah prosedur rute penggabungan, prosedur ini diperlukan untuk menciptakan solusi. Setiap rute diciptakan untuk rute tertutup (siklus Hamiltonian) dengan menetapkan pelanggan

pertama yang dimulai pada depot, dan yang terakhir pelanggan yang kembali ke depot.

Sebaliknya, di OVRP setiap rute diciptakan untuk rute terbuka (jalur Hamiltonian) oleh hanya menempatkan pelanggan pertama yang dimulai pada depot dan tidak mengharuskan untuk kembali ke depot. Akibatnya, dalam prosedur ini, dua solusi OVRP mungkin dibangun, dan kemudian solusi terbaik akan dipilih sebagai solusi OVRP.

2.4.3 Tahap Dua Prosedur Seleksi.

Setelah solusi CW dihasilkan dari daftar *saving list* yang dihasilkan setelah menyortir dalam urutan menurun, daftar *saving list* akan dibuat ulang sebagai yang baru dimana nilai akan dipilih secara acak dan menggunakan *probabilitas*. Pichpibul dan Kawtummachai (2012) memperkenalkan prosedur seleksi dua tahap untuk CVRP. Dalam tulisan ini, kita menyesuaikan prosedur ini untuk menangani dengan OVRP didasarkan pada operasi genetika algoritma. *Saving list* diwakili oleh satu kromosom, dan masing-masing gen mewakili nilai tabungan antara pelanggan i dan j . Pada bagian pertama iterasi, kromosom adalah daftar tabungan diurutkan oleh urutan menurun, tetapi dalam iterasi berikutnya kromosom adalah daftar tabungan berasal dari yang terbaik. T adalah ukuran turnamen yang merupakan nomor acak antara tiga dan enam. Dalam rangka menciptakan roda *roulette*, pemilihan *probabilitas* (p_n) dan *probabilitas* kumulatif (q_n) dengan nilai tabungan (s_n) untuk setiap gen (n) dihitung menggunakan persamaan berikut

$$p_n = \frac{s_n}{\sum_{i \in T} s_i} \text{ for } n \in T$$

$$q_n = \sum_{i \in n} p_i \text{ for } n \in T$$

Setelah itu, selanjutnya dilakukan pemutaran roda dengan nomor acak ($r = 0.38$) dari kisaran antara 0 dan 1. Nilai *saving list* pertama (s_2) akan dipilih untuk menjadi data dari *customers* baru dengan mempertimbangkan r dan q_n . Jika $r \leq q_1$, maka *list* pertama terpilih menjadi values 1; jika tidak, pilih *list* lain yang memiliki values n ($2 \leq n \leq T$) sehingga $q_{n-1} < r \leq q_n$. Daftar yang dipilih dihapus dari

customers dan dilakukan perhitungan untuk iterasi berikutnya. Oleh karena itu, prosedur ini akan dijalankan sampai *customers* terakhir dipilih untuk menjadi data dari *customers* yang baru. Ketika *customers* baru mewakili *saving list* yang dihasilkan, selanjutnya dihitung rute penggabungan dengan prosedur rute terbuka untuk menghasilkan solusi OVRP baru. Selanjutnya membandingkan dua solusi, daftar baru akan menggantikan daftar sebelumnya hanya jika solusi baru lebih baik dari sebelumnya. Kriteria penerimaan ini disebut suatu dasar pencarian lingkungan variabel yang hanya menerima perbaikan. Perhitungan dilanjutkan sampai menghentikan kriteria, yaitu jumlah iterasi global yang memuaskan.

